

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

**BLACK BORDERS**

**IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**

**FADED TEXT OR DRAWING**

**BLURRED OR ILLEGIBLE TEXT OR DRAWING**

**SKEWED/SLANTED IMAGES**

**COLOR OR BLACK AND WHITE PHOTOGRAPHS**

**GRAY SCALE DOCUMENTS**

**LINES OR MARKS ON ORIGINAL DOCUMENT**

**REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**

**OTHER:** \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**

## WEST Search History

[Hide Items](#) | [Restore](#) | [Clear](#) | [Cancel](#)

DATE: Tuesday, September 14, 2004

[Hide?](#) [Set Name Query](#)

**Hit Count**

*DB=PGPB,USPT,USOC; PLUR=YES; OP=ADJ*

<input type="checkbox"/>	L9	(module or subroutine or routine) same (dangling and argument)	3
<input type="checkbox"/>	L8	(module or subroutine or routine) same (dangling and parameter)	9
<input type="checkbox"/>	L7	L6 and reuse	5
<input type="checkbox"/>	L6	L4	38

*DB=EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ*

<input type="checkbox"/>	L5	(module or subroutine or routine) same (dangling and reference)	1
--------------------------	----	---	---

*DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ*

<input type="checkbox"/>	L4	(module or subroutine or routine) same (dangling and reference)	39
<input type="checkbox"/>	L3	(module or subroutine or routine) same (call\$ and reference)	24792

*DB=PGPB,USPT,USOC,PLUR=YES,OP=ADJ*

<input type="checkbox"/>	L2	L1 or 707/205,206.ccls.	3152
<input type="checkbox"/>	L1	717/106-119,162-166.ccls.	2046

END OF SEARCH HISTORY



## THE ACM DIGITAL LIBRARY

[Feedback](#) [Report a problem](#) [Satisfaction survey](#)Terms used **dangling reference NULL reuse**

Found 78 of 142,346

Sort results by relevance

 Save results to a Binder[Try an Advanced Search](#)

Display results expanded form

 [Search Tips](#)[Try this search in The ACM Guide](#) Open results in a new window

Results 1 - 20 of 78

Result page: 1 [2](#) [3](#) [4](#) [next](#)

Relevance scale

**1 The detection of dangling references in C++ programs**

Richard A. Eyre-Todd

March 1993 **ACM Letters on Programming Languages and Systems (LOPLAS)**, Volume 2

Issue 1-4

Full text available: pdf(500.23 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#), [review](#)

The smart pointer is a programming technique for the C++ language that extends the functionality of the simple pointer. Smart pointers have previously been used to support persistence, distributed objects, reference counting, and garbage collection. This article will show how smart pointers can provide an inexpensive method for detecting dangling pointers to dynamic objects that can be added to any standard C++ implementation.

**Keywords:** dangling pointers, smart pointers**2 Region inference for an object-oriented language**

Wei-Ngan Chin, Florin Craciun, Shengchao Qin, Martin Rinard

June 2004 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2004 conference on Programming language design and implementation**, Volume 39 Issue 6Full text available: pdf(207.31 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Region-based memory management offers several important potential advantages over garbage collection, including real-time performance, better data locality, and more efficient use of limited memory. Researchers have advocated the use of regions for functional, imperative, and object-oriented languages. Lexically scoped regions are now a core feature of the Real-Time Specification for Java (RTSJ)[5]. Recent research in region-based programming for Java has focused on region checking, which require ...

**Keywords:** downcasts, memory management, method overriding, object-oriented languages, region inference, type systems**3 Memory safety without runtime checks or garbage collection**

Dinakar Dhurjati, Sumant Kowshik, Vikram Adve, Chris Lattner

June 2003 **ACM SIGPLAN Notices , Proceedings of the 2003 ACM SIGPLAN conference on Language, compiler, and tool for embedded systems**, Volume 38 Issue 7Full text available: pdf(245.47 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Traditional approaches to enforcing memory safety of programs rely heavily on runtime checks of memory accesses and on garbage collection, both of which are unattractive for embedded applications. The long-term goal of our work is to enable 100% static enforcement of memory safety for embedded programs through advanced compiler techniques and minimal semantic restrictions on programs. The key result of this paper is a compiler technique that ensures memory safety of dynamically allocated memory ...

**Keywords:** automatic pool allocation, compilers, embedded systems, programming languages, region management, security, static analysis

**4 Special issue on persistent object systems: Tigukat: a uniform behavioral objectbase management system**

M. Tamer Özsu, Randal Peters, Duane Szafron, Boman Irani, Anna Lipka, Adriana Muñoz  
July 1995 **The VLDB Journal — The International Journal on Very Large Data Bases**,  
Volume 4 Issue 3

Full text available:  pdf(2.78 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

We describe the TIGUKAT objectbase management system, which is under development at the Laboratory for Database Systems Research at the University of Alberta. TIGUKAT has a novel object model, whose identifying characteristics include a purely behavioral semantics and a uniform approach to objects. Everything in the system, including types, classes, collections, behaviors, and functions, as well as meta-information, is a first-class object with well-defined behavior. In this way, the model abstr ...

**Keywords:** database management, objectbase management, persistent storage system, reflective system

**5 Incommunicado: efficient communication for isolates**

Krzysztof Palacz, Jan Vitek, Grzegorz Czajkowski, Laurent Daynas  
November 2002 **ACM SIGPLAN Notices , Proceedings of the 17th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications**, Volume 37 Issue 11

Full text available:  pdf(386.23 KB) Additional Information: [full citation](#), [abstract](#), [references](#)

Executing computations in a single instance of safe language virtual machine can improve performance and overall platform scalability. It also poses various challenges. One of them is providing a fast inter-application communication mechanism. In addition to being efficient, such a mechanism should not violate any functional and non-functional properties of its environment, and should also support enforcement of application-specific security policies. This paper explores the design and implement ...

**Keywords:** application isolation, inter-application communication

**6 Ownership types for safe region-based memory management in real-time Java**

Chandrasekhar Boyapati, Alexandru Salcianu, William Beebe, Martin Rinard  
May 2003 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2003 conference on Programming language design and implementation**, Volume 38 Issue 5

Full text available:  pdf(375.18 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The Real Time Specification for Java (RTSJ) allows a program to create real-time threads with hard real-time constraints. Real-time threads use region-based memory management to avoid unbounded pauses caused by interference from the garbage collector. The RTSJ uses runtime checks to ensure that deleting a region does not create dangling references

and that real-time threads do not access references to objects allocated in the garbage-collected heap. This paper presents a static type system that ...

**Keywords:** encapsulation, ownership types, real-time, regions

7 Minimizing reference count updating with deferred and anchored pointers for functional data structures 

Henry G. Baker

September 1994 **ACM SIGPLAN Notices**, Volume 29 Issue 9

Full text available:  pdf(664.75 KB) Additional Information: [full citation](#), [citations](#), [index terms](#)

8 Portable and efficient dynamic storage management in Ada 

Ron Kownacki, S. Tucker Taft

December 1987 **Proceedings of the 1987 annual ACM SIGAda international conference on Ada**

Full text available:  pdf(875.71 KB) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

---

9 Abstract data types are under full control with Ada 9X 

Magnus Kempe

November 1994 **Proceedings of the conference on TRI-Ada '94**

Full text available:  pdf(1.23 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Ada 83 did not provide enough control on the creation, assignment, and destruction of objects of user-defined types. This lack of control restricted type composition and prevented the full exercise of information hiding for abstract data types. Ada 9X brings new mechanisms supporting the full control of abstract data types, powerful type composition, and more extensive information hiding. With better control of abstract data types, Ada code will be easier to write, understand, maintain, and ...

10 Fast detection of communication patterns in distributed executions 

Thomas Kunz, Michiel F. H. Seuren

November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**

Full text available:  pdf(4.21 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

11 The design of the E programming language 

Joel E. Richardson, Michael J. Carey, Daniel T. Schuh

July 1993 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 15 Issue 3

Full text available:  pdf(2.78 MB) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#), [review](#)

**Keywords:** extensible database systems, persistent object management

**12 A real-time garbage collector based on the lifetimes of objects**

Henry Lieberman, Carl Hewitt

June 1983 **Communications of the ACM**, Volume 26 Issue 6

Full text available:  pdf(1.37 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

In previous heap storage systems, the cost of creating objects and garbage collection is independent of the lifetime of the object. Since objects with short lifetimes account for a large portion of storage use, it is worth optimizing a garbage collector to reclaim storage for these objects more quickly. The garbage collector should spend proportionately less effort reclaiming objects with longer lifetimes. We present a garbage collection algorithm that (1) makes storage for short-li ...

**Keywords:** LISP, algorithms, languages, lisp, object-oriented programming, parallel processing, performance, real-time garbage collection, reference counting, virtual memory

**13 Borrow, copy or steal?: loans and larceny in the orthodox canonical form**

Anthony J. H. Simons

October 1998 **ACM SIGPLAN Notices, Proceedings of the 13th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications**, Volume 33 Issue 10

Full text available:  pdf(2.09 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Dynamic memory management in C++ is complex, especially across the boundaries of library abstract data types. C++ libraries designed in the orthodox canonical form (OCF) alleviate some of the problems by ensuring that classes which manage any kind of heap structures faithfully copy and delete these. However, in certain common circumstances, OCF heap structures are wastefully copied multiple times. General reference counting is not an option in OCF, since a shared body violates the intended value ...

**Keywords:** C++, borrowing, copy-on-write, implementation strategies, larceny, memory management, stealing, transfer of ownership

**14 The building blocks for specifying communication behavior of complex objects: an activity-driven approach**

Ling Liu, Robert Meersman

June 1996 **ACM Transactions on Database Systems (TODS)**, Volume 21 Issue 2

Full text available:  pdf(699.84 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Communication behavior represents dynamic evolution and cooperation of a group of objects in accomplishing a task. It is an important feature in object-oriented systems. We propose the concept of activity as a basic building block for declarative specification of communication behavior in object-oriented database systems, including the temporal ordering of message exchanges within object communication and the behavioral relationships between activity executions. We formally introduce two ki ...

**Keywords:** activity aggregation, activity patterns, activity specialization, communication behavior, first-order temporal logic, object-oriented databases, synchronization schemes

**15**

**Using model dataflow graphs to reduce the storage requirements of constraints**

Bradley T. Vander Zanden, Richard Halterman

September 2001 **ACM Transactions on Computer-Human Interaction (TOCHI)**, Volume 8

Issue 3

Full text available:  pdf(1.28 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Dataflow constraints allow programmers to easily specify relationships among application objects in a natural, declarative manner. Most constraint solvers represent these dataflow relationships as directed edges in a dataflow graph. Unfortunately, dataflow graphs require a great deal of storage. Consequently, an application with a large number of constraints can get pushed into virtual memory, and performance degrades in interactive applications. Our solution is based on the observation that obj ...

**Keywords:** Class-instance model, dataflow constraints, graphical interfaces, language design and implementation, programming environments, prototype-instance model, storage optimization

## 16 Language support for regions

David Gay, Alex Aiken

May 2001 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2001 conference on Programming language design and implementation**, Volume 36 Issue 5

Full-text available:  pdf(1.44 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Region-based memory management systems structure memory by grouping objects in regions under program control. Memory is reclaimed by deleting regions, freeing all objects stored therein. Our compiler for C with regions, RC, prevents unsafe region deletions by keeping a count of references to each region. Using type annotations that make the structure of a program's regions more explicit, we reduce the overhead of reference counting from a maximum of 27% to a maximum of 11% on a suite of reali ...

## 17 Toward a Dexter-based model for open hypermedia: unifying embedded references and link objects

Kaj Grønbæk, Randall H. Trigg

March 1996 **Proceedings of the the seventh ACM conference on Hypertext**

Full text available:  pdf(1.31 MB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

**Keywords:** Dexter hypertext reference model, dynamic hypermedia, embedded links, generic links, link objects, open hypermedia

## 18 Checkmate: cornering C++ dynamic memory errors with checked pointers

Scott M. Pike, Bruce W. Weide, Joseph E. Hollingsworth

March 2000 **ACM SIGCSE Bulletin , Proceedings of the thirty-first SIGCSE technical symposium on Computer science education**, Volume 32 Issue 1

Full text available:  pdf(554.01 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Pointer errors are stumbling blocks for student and veteran programmers alike. Although languages such as Java use references to protect programmers from pointer pitfalls, the use of garbage collection dictates that languages like C++ will still be used for real-time mission-critical applications. Pointers will stay in the classroom as long as they're used in industry, so as educators, we must find better ways to teach them. This paper presents checked pointers, a simple wr ...

## 19 GENOA—a customizable, front-end-retargetable source code analysis framework

Premkumar T. Devanbu

April 1999 **ACM Transactions on Software Engineering and Methodology (TOSEM)**,

Volume 8 Issue 2

Full text available:  pdf(241.27 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Code analysis tools provide support for such software engineering tasks as program understanding, software metrics, testing, and reengineering. In this article we describe GENOA, the framework underlying application generators such as Aria and GEN++ which have been used to generate a wide range of practical code analysis tools. This experience illustrates front-end retargetability of GENOA; we describe the features of the GENOA framework that allow it to be ...

**Keywords:** code inspection, metrics, reverse engineering, source analysis

## 20 Shoring up persistent applications

Michael J. Carey, David J. DeWitt, Michael J. Franklin, Nancy E. Hall, Mark L. McAuliffe, Jeffrey F. Naughton, Daniel T. Schuh, Marvin H. Solomon, C. K. Tan, Odysseas G. Tsatalos, Seth J. White, Michael J. Zwilling

May 1994 **ACM SIGMOD Record , Proceedings of the 1994 ACM SIGMOD international conference on Management of data**, Volume 23 Issue 2

Full text available:  pdf(140 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

SHORE (Scalable Heterogeneous Object REpository) is a persistent object system under development at the University of Wisconsin. SHORE represents a merger of object-oriented database and file system technologies. In this paper we give the goals and motivation for SHORE, and describe how SHORE provides features of both technologies. We also describe some novel aspects of the SHORE architecture, including a symmetric peer-to-peer server architecture, server customization through an extensible ...

Results 1 - 20 of 78

Result page: [1](#) [2](#) [3](#) [4](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2004 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)

Help FAQ Terms IEEE Peer Review

## Quick Links

## Welcome to IEEE Xplore®

- Home
- What Can I Access?
- Log-out

## Tables of Contents

- Journals & Magazines
- Conference Proceedings
- Standards

## Search

- By Author
- Basic
- Advanced

## Member Services

- Join IEEE
- Establish IEEE Web Account
- Access the IEEE Member Digital Library

## IEEE Xplore Digital Library

- Access the IEEE Enterprise File Cabinet

## Print Format

Your search matched **4** of **1071730** documents.A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance** in **Descending** order.

## Refine This Search:

You may refine your search by editing the current search expression or entering a new one in the text box.

dangling&lt;and&gt;reference

Search

 Check to search within this result set

## Results Key:

JNL = Journal or Magazine CNF = Conference STD = Standard

**1 Hardware-based solution detecting illegal references in real-time Java***Higuera-Toledano, M.T.;*

Real-Time Systems, 2003. Proceedings. 15th Euromicro Conference on , 2-4 July 2003

Pages:229 - 337

[Abstract] [PDF Full-Text (367 KB)] IEEE CNF

**2 Complete distributed garbage collection: an experience with Rotor***Veiga, L.; Ferreira, P.;*

Software, IEE Proceedings- [see also Software Engineering, IEE Proceedings] , Volume: 150 , Issue: 5 , 27 Oct. 2003

Pages:283 - 290

[Abstract] [PDF Full-Text (277 KB)] IEE JNL

**3 Automatic link generation and repair mechanism for document management***Shimada, T.; Futakata, A.;*

System Sciences, 1998., Proceedings of the Thirty-First Hawaii International Conference on , Volume: 2 , 6-9 Jan. 1998

Pages:226 - 235 vol.2

[Abstract] [PDF Full-Text (808 KB)] IEEE CNF

**4 Dynamic detection of access errors and illegal references in RTSJ***Higuera-Toledano, T.M.; de Miguel-Cabello, M.A.;*

Real-Time and Embedded Technology and Applications Symposium, 2002. Proceedings. Eighth IEEE , 24-27 Sept. 2002

Pages:101 - 110

[\[Abstract\]](#) [\[PDF Full-Text \(554 KB\)\]](#) [IEEE CNF](#)

---

[Home](#) | [Log-out](#) | [Journals](#) | [Conference Proceedings](#) | [Standards](#) | [Search by Author](#) | [Basic Search](#) | [Advanced Search](#) | [Join IEEE](#) | [Web Account](#) | [New this week](#) | [OPAC Linking Information](#) | [Your Feedback](#) | [Technical Support](#) | [Email Alerting](#) | [No Robots Please](#) | [Release Notes](#) | [IEEE Online Publications](#) | [Help](#) | [FAQ](#) | [Terms](#) | [Back to Top](#)

Copyright © 2004 IEEE — All rights reserved



dangling reference" NULL module

Search

Shortcuts Advanced Search Preferences

## Search Results

Results 1 - 20 of about 350 for "dangling reference" NULL module. Search took 0.53 seconds. ([About this page...](#)1. [Appendix B: XPCOM API Reference](#)

... a non-owning, non-dangling **reference**. Clients own the ... Note: If ptr is **null**, this function behaves like ... This is in [www.mozilla.org/projects/xpcom/book/cxch/html/appB.html](http://www.mozilla.org/projects/xpcom/book/cxch/html/appB.html) - 291k - [Cached](#) - [More pages from this site](#)

2. [PyCXX: Writing Python Extensions in C++](#)

PyCXX: Writing Python Extensions in C++ Barry Scott, [barry@barrys-emacs.org](mailto:barry@barrys-emacs.org). Paul F. Dubois, [dubois1@llnl.gov](mailto:dubois1@llnl.gov). you access to the releases, the CVS repository, and more. [cxx.sourceforge.net/](http://cxx.sourceforge.net/) - 56k - [Cached](#) - [More pages from this site](#)

3. [Writing Python Extensions in C++](#)

Writing Python Extensions in C++ Paul F. Dubois, [dubois1@llnl.gov](mailto:dubois1@llnl.gov). Lawrence Livermore National Laboratory. CXX [w3.pppl.gov/~hammett/comp/python/LLNLDistribution11/CXX/Doc/cxx.htm](http://w3.pppl.gov/~hammett/comp/python/LLNLDistribution11/CXX/Doc/cxx.htm) - [More pages from this site](#)

4. [Writing Python Extensions in C++](#)

... New(); if (!d) { return **NULL**; } f = PyInt\_NEW(k+1) ... When writing an extension **module** method, you can use the [www.atd.ucar.edu/software/CXX](http://www.atd.ucar.edu/software/CXX) - 68k - [Cached](#) - [More pages from this site](#)

5. [1 \(PDF\)](#)

... of the discussion of **module**-based programming enumerations modules for ... errors, e.g. **dangling reference** pro **NULL**; int i = 2 ...

[www.mathcs.sjsu.edu/faculty/nguyend/cs140/2003fall/slides/muldner/chap\\_08\\_3.pdf](http://www.mathcs.sjsu.edu/faculty/nguyend/cs140/2003fall/slides/muldner/chap_08_3.pdf) - 193k - [View as html](#) - [More pages from this site](#)

6. <http://www.towson.edu/users/hendrick/LECNOTES.txt>

COSC 237 Lecture Notes Prof. C. Dierbach Dept. of Computer and Information Sciences Towson State University S itself. Such input/output ... some data object – **null** value The **null** value is simply ... delete intptr1 }; c. **Dangling Re** [www.towson.edu/users/hendrick/LECNOTES.txt](http://www.towson.edu/users/hendrick/LECNOTES.txt) - 198k - [Cached](#) - [More pages from this site](#)

7. [A Tutorial on MC++, the Specification Language for the NUMLAB System](#)

... input "inp" will remain with a **dangling reference** to A's T&starf#star; destroyed ... let a **module** share some data [www.win.tue.nl/~maubach/numlab/documentation/reference-guide/mcdoc/mcd\\_oc.html](http://www.win.tue.nl/~maubach/numlab/documentation/reference-guide/mcdoc/mcd_oc.html) - 119k - [Cached](#) - [More pages from this site](#)

8. <http://www.win.tue.nl/~maubach/numlab/documentation/reference-guide/mcdoc.tex-original>

Further, the process of developing {sc \NumLabNetEdit} libraries is explained and detailed with various design cons [www.win.tue.nl/~maubach/numlab/documentation/reference-guide/mcdoc.tex](http://www.win.tue.nl/~maubach/numlab/documentation/reference-guide/mcdoc.tex) - origin... - 97k - [Cached](#) - [More pages from this site](#)

9. [1 \(PDF\)](#)

... if((i = malloc(sizeof(int)) == **NULL**){return **NULL**; } i = p ... Company, 20008: The **Dangling Reference** ProblemSta Tokenizing ...

[www.cs.helsinki.fi/u/gurtov/c02/lect5.pdf](http://www.cs.helsinki.fi/u/gurtov/c02/lect5.pdf) - 290k - [View as html](#) - [More pages from this site](#)

10. <http://www.omg.org/docs/ptc/03-03-09.txt>

2 Minor IDL Changes: Those IDL changes that made very minor changes or enhancements (changing names, data [www.omg.org/docs/ptc/03-03-09.txt](http://www.omg.org/docs/ptc/03-03-09.txt) - 35k - [Cached](#) - [More pages from this site](#)

11. <http://kerneljanitors.org/TODO>

... checking for **NULL** on probe routines for net drivers - convert ... failure for **init\_module**, the **module** is unloaded b

12. [Chapter 1: Introduction](#) ↗

... Chapter 7: Functions, Scope, and Introduction to Module-based Programming ... 8: Generic Pointers and NULL.  
Overriding ...  
[dragon.acadiau.ca/~jdiamond/comp1033/slides/Slides-in-html](http://dragon.acadiau.ca/~jdiamond/comp1033/slides/Slides-in-html) - 32k - [Cached](#) - [More pages from this site](#)

13. <http://downloads.activestate.com/pub/staff/gsar/APC/perl-5.8.x/pod/perldiag.pod> ↗

... routines was passed a null HV pointer. =item Bad index ... to create a dangling reference. =item Did not produce  
[downloads.activestate.com/pub/staff/gsar/APC/perl-5.8.x/pod/perldiag.pod](http://downloads.activestate.com/pub/staff/gsar/APC/perl-5.8.x/pod/perldiag.pod) - 165k - [Cached](#) - [More pages from this site](#)

14. [Idioms of Flare](#) ↗

... class defined in the module "people", then it knows to ... those references to be set to null. Dynamically deleting  
[www.flare.org/idioms.html](http://www.flare.org/idioms.html) - 51k - [Cached](#) - [More pages from this site](#)

15. [C](#) ↗

Tips and Helpful Hints for C and C++ ... general: It might be NULL or it might be a ... or deleting a non-NULL pointer  
[techsinfo.freewebsitethosting.com/programing/c/c.htm](http://techsinfo.freewebsitethosting.com/programing/c/c.htm) - 394k - [Cached](#) - [More pages from this site](#)

16. [the current todo list](#) ↗

... checking for NULL on probe routines for net drivers - convert ... failure for init\_module, the module is unloaded b  
[www.kerneljanitors.org/TODO](http://www.kerneljanitors.org/TODO) - 25k - [Cached](#) - [More pages from this site](#)

17. [ActivePerl Help - Online Docs : perldiag](#) ↗

... the internal hash routines was passed a null HV pointer ... automatically loads the Errno.pm module. The Errno.pm reference ...  
[aspn.activestate.com/ASPNet/docs/ActivePerl/lib/Pod/perldiag.html](http://aspn.activestate.com/ASPNet/docs/ActivePerl/lib/Pod/perldiag.html) - 346k - [Cached](#) - [More pages from this site](#)

18. <http://janitor.kernelnewbies.org/TODO> ↗

... checking for NULL on probe routines for net drivers - convert ... failure for init\_module, the module is unloaded b  
[janitor.kernelnewbies.org/TODO](http://janitor.kernelnewbies.org/TODO) - 25k - [Cached](#) - [More pages from this site](#)

19. [NRC Ada & Beyond \(Part 6/6\)](#) ↗

... information hiding through a module construct called a "package" ... initialized to null, and checked for null prior to  
[sw-eng.falls-church.va.us/nrc/nrc6.html](http://sw-eng.falls-church.va.us/nrc/nrc6.html) - 100k - [Cached](#) - [More pages from this site](#)

20. <http://www.cs.helsinki.fi/u/gurtov/c02/clect6.ppt> (MICROSOFT POWERPOINT) ↗

... sizeof(char))) == NULL) return NULL; ... 10: Arrays and Dangling Reference. char \*setName(int i) { ... a call to th  
[www.cs.helsinki.fi/u/gurtov/c02/clect6.ppt](http://www.cs.helsinki.fi/u/gurtov/c02/clect6.ppt) - 367k - [View as html](#) - [More pages from this site](#)

Results Page:

1 [2](#) [3](#) [4](#) [5](#) ► [Next](#)

---

[Web](#) | [Images](#) | [Directory](#) | [Yellow Pages](#) | [News](#) | [Products](#)

Your Search:

Help us improve your search experience. [Send us feedback](#).

[One-click to Mail, Search and More! - Yahoo! Toolbar](#)

Copyright © 2004 Yahoo! Inc. All rights reserved. [Privacy Policy](#) - [Terms of Service](#) - [Submit Your Site](#)

Searching for dangling reference and null and module.

Restrict to: Header Title Order by: Expected citations Hubs Usage Date Try: Google (CiteSeer) Google (Web)  
CSB DBLP

No documents match Boolean query. Trying non-Boolean relevance query.

500 documents found. Order: relevance to query.

Learning Planning Operators by Observation and Practice - Wang (1994) (Correct) (15 citations)

plan repair with possibly incorrect operators. References Anzai, Y. and Simon, H. A. 1979. The Theory and practice in planning include the observation module, the learning module, the planning module, and include the observation module, the learning module, the planning module, and the plan execution  
[www.rpal.rockwell.com/~mei/aips94.ps](http://www.rpal.rockwell.com/~mei/aips94.ps)

Correction of a Memory Management Method for Lock-Free Data.. - Michael, Scott (1995) (Correct) (7 citations)

addresses these problems. The method associates a reference count with each node of reusable memory. A node NEW(1 loop 2 p /SAFEREAD(Freelist) 3 if p =NULL 4-error-out-of-memory 5 if-CAS(Freelist, p,  
[hypatia.dcs.qmw.ac.uk/data/edu/cs.rochester.edu/systems/95.tr599.Memory\\_management\\_for\\_lock-free\\_data\\_structures.ps.gz](http://hypatia.dcs.qmw.ac.uk/data/edu/cs.rochester.edu/systems/95.tr599.Memory_management_for_lock-free_data_structures.ps.gz)

Development, Learning and Evolution in Animats - Kodjabachian, Meyer (1994) (Correct) (2 citations)

process, a cell executes the instruction referenced by the symbol it reads and moves its reading

separated by a comma (no connection is made). Modules can be created by grouping neurons or other

can be created by grouping neurons or other modules between square brackets (two adjoining

[www.biologie.ens.fr/fr/animatlab/perso/kodjaba/jkjamperac.ps.gz](http://www.biologie.ens.fr/fr/animatlab/perso/kodjaba/jkjamperac.ps.gz)

[Recommendation X-904 - Basic Reference (1994)] (Correct)

SC21-N9035-[Recommendation X-904 - Basic Reference Model of Open Distributed Processing -Part 4:

effect is the instantaneous change in state (or the null change) of the object with which that action is through a hierarchically structured set of module instances, communicating in an asynchronous

[ftp.gte.com/pub/odp/1994/part4\\_p2.ps.gz](http://ftp.gte.com/pub/odp/1994/part4_p2.ps.gz)

Mechanisms and Interfaces for Software-Extended Coherent Shared.. - Chaiken (1994) (Correct) (4 citations)

and protocols that adapt dynamically to memory reference patterns. Keywords: multiprocessor, parallel and Dan were responsible for the Sparcle simulator modules of ASIM and NWO. Kirk Johnson and Chris Metcalf partitioning memory and processors into separate modules, these architectures distribute memory  
[ftp.cag.lcs.mit.edu/pub/papers/chaiken-dissert-1-10.ps.Z](http://ftp.cag.lcs.mit.edu/pub/papers/chaiken-dissert-1-10.ps.Z)

Intelligent Computing About Complex Dynamical Systems - Zhao (1994) (Correct)

should change values. A synthesized control reference trajectory consists of a sequence of trajectory  
[www.cis.ohio-state.edu/insight/papers/mcs.ps](http://www.cis.ohio-state.edu/insight/papers/mcs.ps)

A Supersymmetry Approach To Poisson Structures Over.. - Krasilshchik (1995) (Correct)

E E ffl \Delta =0 `E !0 formulae References [10, 12, 13, 15] 1, 6, 7, 11] 3, 4] Consider vector fields on N is denoted, Gamma(is the module of sections for and j k (f) denotes the k-jet -z p times while q h (E 1 is the module of horizontal q-forms. 3. The C-spectral  
[cirm.univ-mrs.fr/pub/EMIS/proceedings/6ICDGA/IV/krasil.ps](http://cirm.univ-mrs.fr/pub/EMIS/proceedings/6ICDGA/IV/krasil.ps)

Nozomi - A Fast, Memory-Efficient Stack Decoder for LVCSR - Schuster (1996) (Correct)

hypothesis for the current LM state and future reference. Compared to the generation of the first best happen to have any extension words, indicated by a NULL pointer on level n \Gamma 1, the next non-NUL on the description of some of the decoder modules and issues, which were found to be important for  
[www.aist-nara.ac.jp/IS/Shikano-lab/staff/1996/mike-s/papers/icslp98.ps.gz](http://www.aist-nara.ac.jp/IS/Shikano-lab/staff/1996/mike-s/papers/icslp98.ps.gz)

Patterns for Reducing Locking Overhead in Multi-threaded.. - Harrison, Schmidt (1996) (Correct)

static Singleton instance method to retrieve a reference to the Singleton before performing operations, Singleton \*instance (void) if (instance\_ NULL) instance\_ new Singleton return instance\_  
[oop.rosweb.ru/cpp/cpptools/TSS-pattern.ps.gz](http://oop.rosweb.ru/cpp/cpptools/TSS-pattern.ps.gz)

Interactive Modular Programming in Scheme - Tung (1992) (Correct) (3 citations)  
by other **modules**, we can simply export a "reference procedure,e.g.`lambda (x )and an`  
Abstract This paper presents a **module** system and a programming environment designed to  
interactive program development in Scheme. The **module** system extends lexical scoping while maintaining  
<ftp://cs.indiana.edu/indra/scheme-repository/doc/pubs/imp.ps.gz>

Towards 3-D model-based tracking and recognition of human.. - Gavrila, Davis (1995) (Correct)  
the matching of a test sequence with several **reference** sequences representing prototypical  
model, the system consists of independent running **modules**, receiving and passing data through their  
[www.umiacs.umd.edu/users/gavrila/iwafgr.ps.Z](http://www.umiacs.umd.edu/users/gavrila/iwafgr.ps.Z)

Segregatory Coordination and Ellipsis in Text Generation - Shaw (1998) (Correct) (6 citations)  
Foundation) and NSF Grants GER-90-2406. References Charles B. Callaway and James C. Lester.  
the architecture of our generation system and the **modules** that handle coordination construction. A  
Aggregation in Sentence PlannER)as the first **module** in the tactical component to handle clause  
[www.cs.columbia.edu/~shaw/papers/colingacl98.ps.gz](http://www.cs.columbia.edu/~shaw/papers/colingacl98.ps.gz)

A Meta-theory for Structured Presentations in the COC - Shulman (1997) (Correct)  
the ability to discharge a presentation **reference** with consistent extensions of a presentation,  
to other theories, starting with the empty or null theory. New theories and their relationships are  
development for successive decomposition of ML **modules** until an executable program is reached. The  
[cse.ogi.edu/pub/tech-reports/1997/97-TH-001.ps.gz](http://cse.ogi.edu/pub/tech-reports/1997/97-TH-001.ps.gz)

36 Problems for Semantic Interpretation - Scheler (Correct)  
predicates mass nouns "lazy" pronouns intensional reference intensional coreference non-compositionality of  
[flop.informatik.tu-muenchen.de/pub/fki/fki-179-93.ps.gz](http://flop.informatik.tu-muenchen.de/pub/fki/fki-179-93.ps.gz)

Using PVM 3.0 to Run Grand Challenge Applications on... - Dongarra, Geist, (1992) (Correct)  
and careful use of locks. PVM 3.0 contains a **reference** port to the Intel iPSC/860 Integration into  
[ftp.netlib.org/ncwn/siam93-pvrgc.ps](http://ftp.netlib.org/ncwn/siam93-pvrgc.ps)

Chase and Axioms for PC Queries and Dependencies - Popa, Tannen (1998) (Correct)  
Q 0 that we wanted. This proves lemma 8.5 (1) References Abi83] S. Abiteboul. Algebraic analogues to  
 $i \ j \ E : A \ j \ sng \ E \ j \ Loop [ff] \ x \ 2 \ E \ 1 \ )E \ 2 \ (x \ j \ null [ff] \ j \ domE \ j \ E \ 1 \ !E \ 2 \ j \ key \ x \ in \ E \ 1 \ )E \ 2 \ (x$   
[www.cis.upenn.edu/~val/tr-MS-CIS-98-34.ps](http://www.cis.upenn.edu/~val/tr-MS-CIS-98-34.ps)

Automatic Memory Management in a Static Type System with . . . - Bonachea, Hurwitz, McPeak (Correct)  
memory management and statically prevents **dangling references** and memory leaks with very low  
management and statically prevents **dangling references** and memory leaks with very low runtime  
[www.cs.berkeley.edu/~bonachea/263/paper.ps](http://www.cs.berkeley.edu/~bonachea/263/paper.ps)

A Hypertext System for Integrating Heterogeneous, Autonomous.. - Noll, Scacchi (1994) (Correct) (2 citations)  
storing dependency relationships, function cross **references** (tags"and versions. Thus, the global  
object comparability are problems: one person's **module** is another 's subsystem dependency  
style communica- GetObject(dht.c"SELECT \*FROM modules WHERE name =dht.c" Gateway Local Repository  
[cwis.usc.edu/dept/ATRIUM/Papers/Integrating\\_Software.Repositories.ps](http://cwis.usc.edu/dept/ATRIUM/Papers/Integrating_Software.Repositories.ps)

Towards Alias-Free Pointers - Naftaly Minsky (1996) (Correct) (38 citations)  
to it, left anywhere in the system, becomes a **dangling reference**, which may cause serious errors that  
2 But if c is a dynamic object, addressed via a **reference** variable p c contained in x then, the scope  
first, the value of u is copied into v, then u is **nullified** i.e.the value void (the **null** pointer of  
[www.cs.rutgers.edu/~minsky/papers/unique-paper.ps](http://www.cs.rutgers.edu/~minsky/papers/unique-paper.ps)

CSDC - The MoTiV Car Speech Data Collection - Langmann, Pfitzinger.. (1998) (Correct) (5 citations)  
under traffic conditions of (factory-trained) **references** that have been trained from "clean" speech  
[www.phonetik.uni-muenchen.de/Publications/Pfitzinger\\_LREC98b.ps](http://www.phonetik.uni-muenchen.de/Publications/Pfitzinger_LREC98b.ps)

First 20 documents Next 20

Try your query at: [Google \(CiteSeer\)](#) [Google \(Web\)](#) [CSB](#) [DBLP](#)

